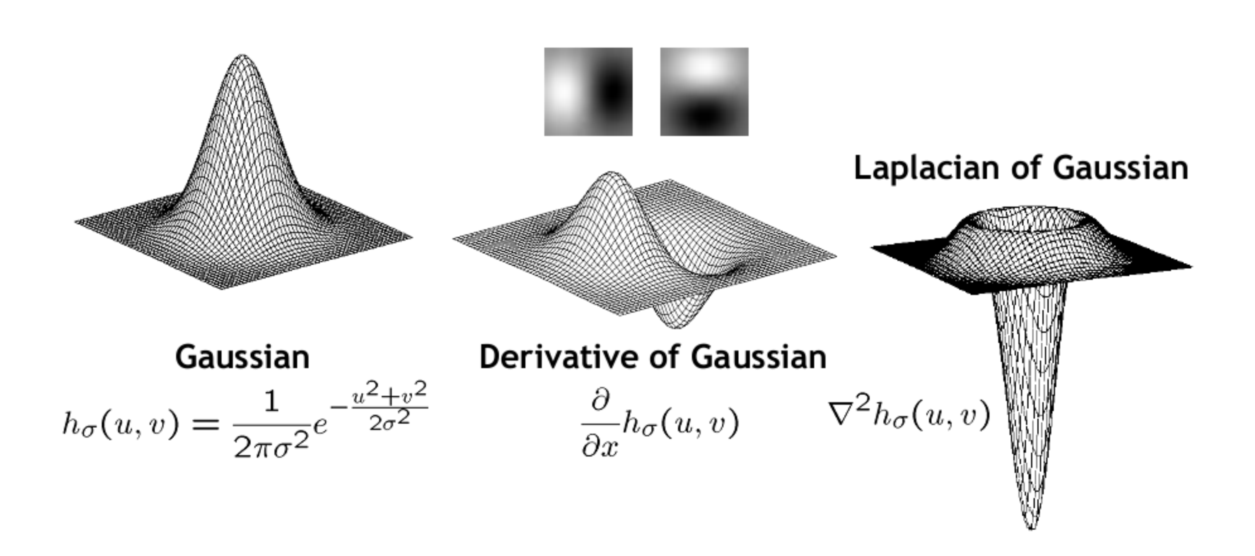
جامعة دمشق
كلية الهندسة المعلوماتية
قسم الذكاء الصنعي



المنطق الضبابي

في معالجة الصور

إعداد: محمد وسيم ابوزينة



ازالة الضجيج باستخدام المنطق الضبابى :

في السنوات الماضية دخلت التقنيات الضبابية مجال الرؤية الحاسوبية ومعالجة الصور بقوة وامست تنافس الطرق الكلاسيكية من حيث الجودة والاداء ، ونذكر بالتحديد الفلاتر الغير خطية لمعالجة الصور. العديد من الطرق والمنهجيات تم العمل عليها وطرحها حتى الآن . كما تتبين اهمية وقوة الاستدلال الضبابي في اوجها بحالات عدم اليقين uncertainty التي تؤثر على عملية استخراج المعلومات من المعطيات المخربة بالضجيج في الصور على سبيل المثال لا الحصر .

معاملات الاستدلال الضبابي المقادة بالقواعد If Else action هي معاملات غير خطية بحيث تتبنى قواعد ضبابية fuzzy rule لمعالجة معطيات الصور . يوجد العديد من المعاملات operators الضبابية المتوفرة لحل العديد من مشاكل معالجة الصور بما في ذلك تنعيم الصورة somoothing ، استخراج الحواف ، شحذ الصور Sharpening . الخ .....

يوجد صنف جديد من معاملات الاستدلال الضبابي التي تتعامل مع ازالة وحذف الضجيج من الصور يدعى الفلتر PWL-Fire هذا الفلتر مبني على piecewise linear fuzzy بحيث يتغير شكل هذا الفلتر بشكل ديناميكي بحيث يعتمد على خصائص الصورة المحلية بالاعتماد على هذا التصميم يمكن تقديم طريقة فعالة لإزالة الضجيج بحيث لا تؤثر سلبا على دقة وجودة الصورة .

الفلتر الذي سنتعامل معه يعتمد بشكل اساسي على معالجة نافذة جوارات البكسل Window-Based بحيث نجد اننا لا نعالج فقط كل بكسل لوحدة وانما نعالج ايضاً مجموعة البكسلات المجاورة الثمانية حسب حجم النافذة التي نتعامل معها وعندما يختلف حجم النافذة تختلف القواعد التي سنتعامل معها ولكن الحجم 3*3 هو الحجم المثالي للعمل وهو ما سنعتمد عليه في الطريقة المطروحة .

تقوم عملية المعالجة على تطبيق قواعد ضبابية لتقدير معدل التصحيح اللازم اجراؤه وفق قواعد من الشكل if then else

وفي حال تحققت احدى القواعد نقوم بعملية تصحيح قيمة البكسل اللونية الرمادية Luminance بينما لا نقوم باي تعديل في حال عدم تحقق اي من القواعد .

لنفرض على سبيل المثال اننا امام صورة X ويكون X(n) هو قيمة البكسل اللونية الرمادية في المكان n في الصورة المميزة بالضجيج ليكن لدينا W(n) الذي يمثل مجموعة من بكسلات الجوار للبكسل المعالج بحيث نجد ان البكسلات المجاورة هي البكسلات التي تقع ضمن النافذة 3*3 حول البكسل المعالج X(n) .

W(n)= { Xj(n) ;   j=1,2,3,4,5,6,7,8 }

يمكن تعريف متحولات الدخل للمعامل بانها الفروق بين تدرجات الرمادي luminance differences .

Xj (n) = Xj (n) – X(n)

ان متحول الخرج y(n) يمثل التعديل اللازم القيام به واضافة هذه القيمة الى قيمة لون البكسل الاصلية X(n) ، الامر الذي يؤدي الى قيمة تدرج رمادي جديدة ممثلة بالمعادلة :

Y(n) = X(n) + Y(n) old

باستعمال القواعد الضبابية يقوم المعامل اللاخطي بالمطابقة maping بين مجموعة متحولات الدخل ومتحول الخرج ، بحيث ينتج بالنهاية معامل التعديل الذي يقوم بحذف اشارة الضجيج .

كي نقوم بزيادة فعالية هذه الطريقة نقوم بتطبيق الفلتر بشكل عودي على معطيات الصورة بحيث تعطي قيمة y(n) الجديدة قيمة التدرج الرمادي للبكسل المعالج X(n) في نهاية المعالجة .

الشكل التالي يبين نافذة 3*3

X_1	X_2	X_3
X_8	X	X_4
X_7	X_6	X_5

Fig. 1: 3x3 window

عندما نتعامل مع L مستوى تدرج رمادي فإن القيم التي تاخذها متحولات الدخل والخرج تكون ضمن المجال [-L+1, L-1]

بالنسبة للمجموعات الضبابية التي تمثل دخل للعملية فإننا نقوم بتعريف مجموعتين ضبابيتين LP,LN .
large positive , large Negative . ان قيمة تابع الانتماء mLN للمجموعة الضبابية LN يمثل بكل بساطة بالشكل التالي :

mLN(u) = mLP(-u)

بينما تاخذ المجموعة الضبابية mLP الشكل 2 :
تتعامل القواعد الضبابية مع العديد من الاشكال والنماذج المختلفة للنافذة كي تقوم بالكشف عن الضجيج . على سبيل المثال يمكن اعتبار النموذج التالي المشكل من تدرجات لونية ل 3 بكسلات x2,x4,x6 . بالتالي نجد ان الفروق بين التدرجات اللونية سيتم تقيمها وفق التالي x2=x2-x, x4=x4-x,x6=x6-x . بسبب التناظرية في عملية المعالجة فإنه يمكننا توليد زوج من القواعد الضبابية بحيث تكون معرفة بالشكل التالي :

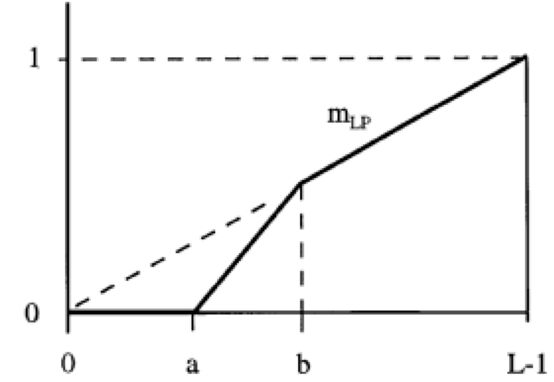


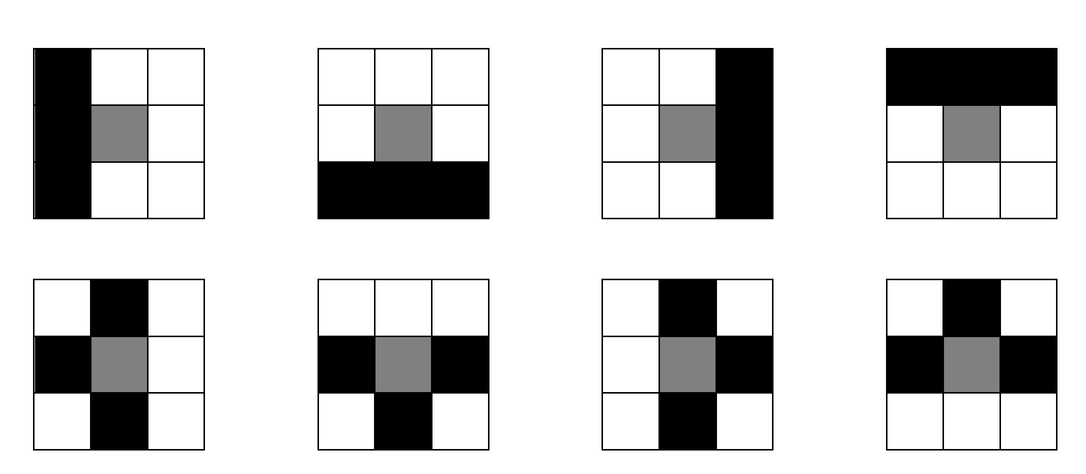
Fig. 2: Piecewise linear fuzzy set LP.

IF (•x_2, LP) AND (•x_4, LP) AND (•x_6, LP)
THEN (•y,PO);

IF (•x_2,LN) AND (•x_4,LN) AND (•x_6,LN)
THEN (•y,NE);

تعني PO موجب positive و تعني NE سلبي negative التي تمثل singeltons متمركزة حول L-1 و L+1– .

تم استعمال القواعد التالية على النافذة :



يجدر الذكر بان هذه القواعد السابقة هي مصممة لمعالجة موجات الضجيج السالبة والموجبة ،

بالنسبة للخرج يتم تقيمه رقمياً وفق التالي :

New Luminance = Old Luminance + Variance

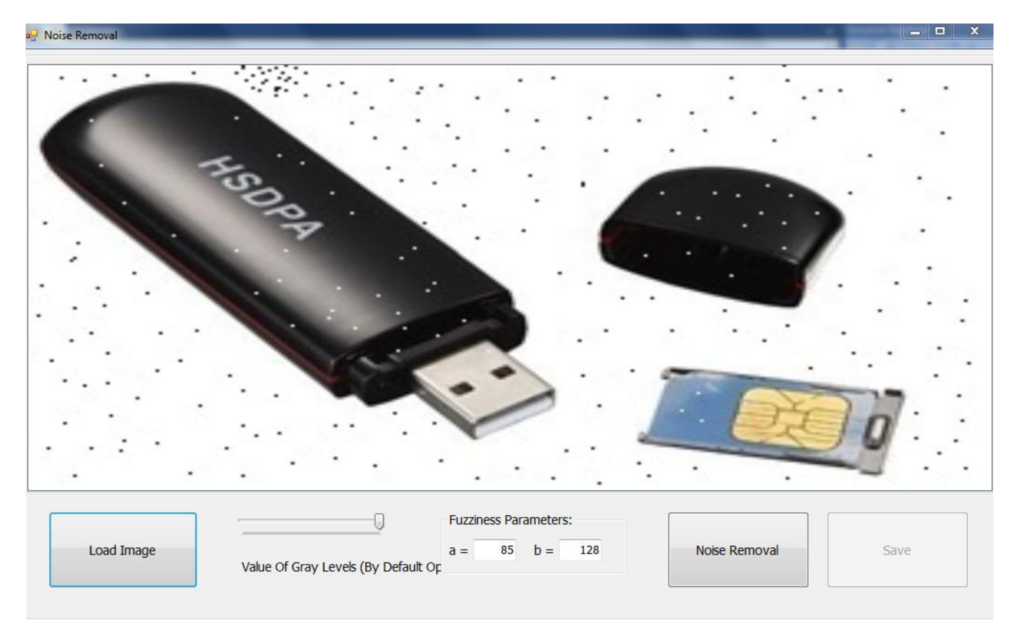
Variance = (L-1) (λ1 – λ2) :

λ1 = Max { Min { mLP (ΔXj) : j ε Ai} , I = 1 .. N}

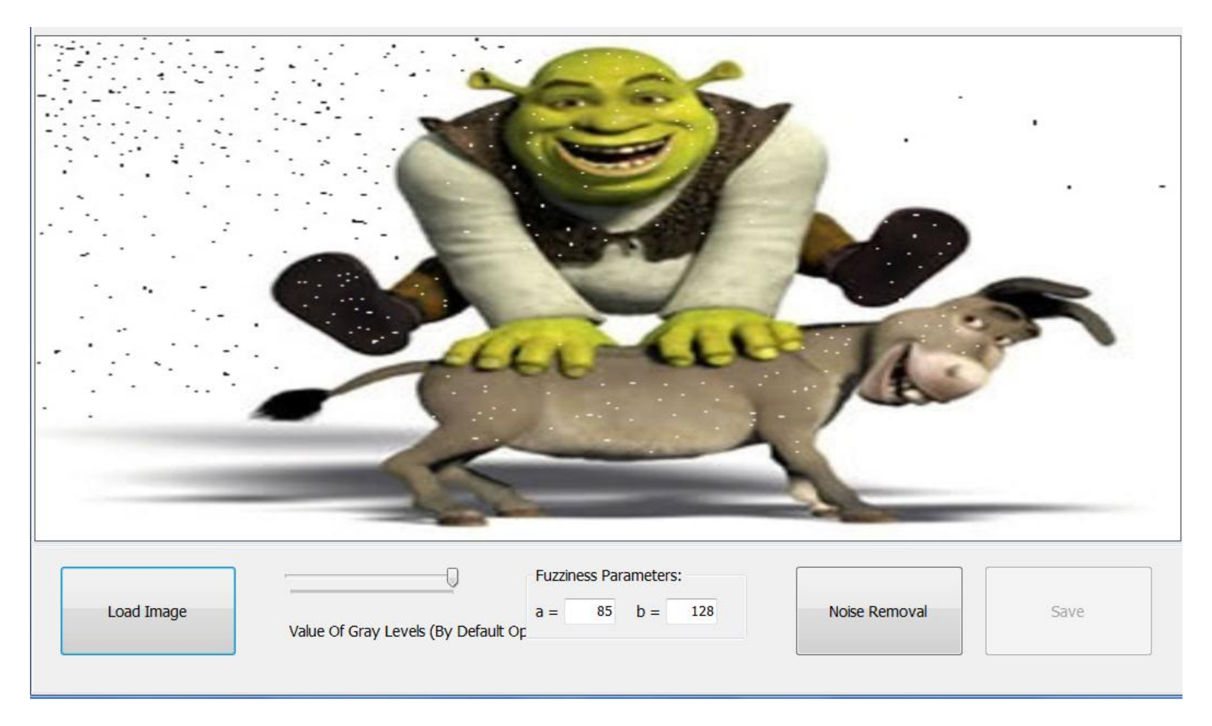
λ2 = Max { Min { mLN (ΔXj) : j ε Ai} , I = 1 .. N}

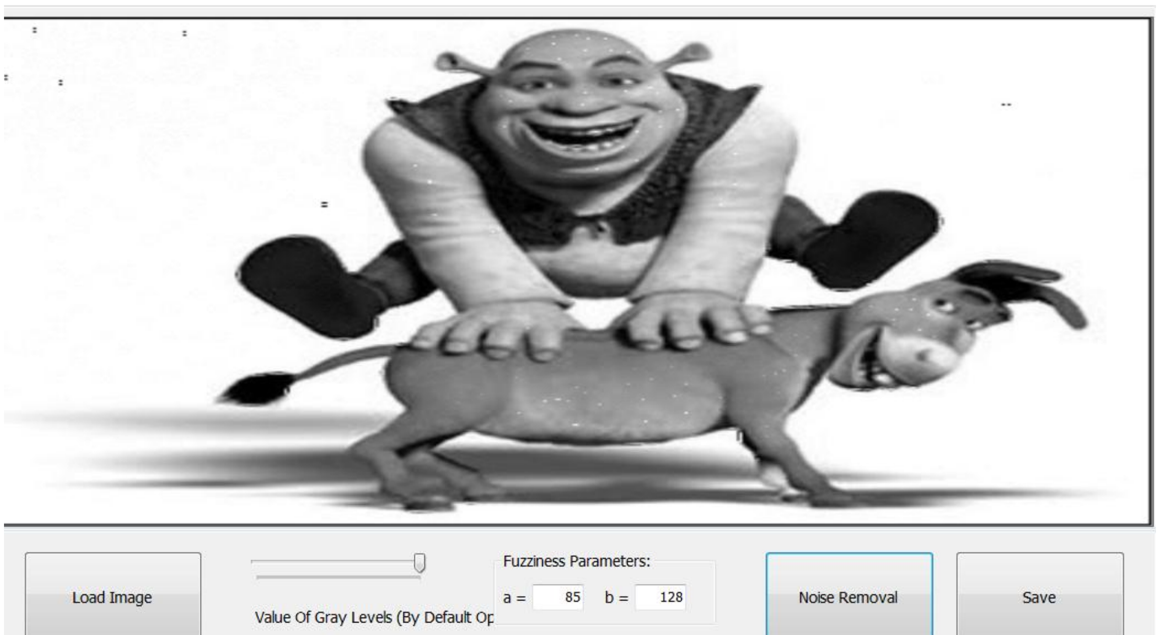
ان آلية حفظ التفاصيل للفلتر تعتمد بشكل رئيسي على اختيار براميترات المجموعة الضبابية a,b . بالتاكيد فإن تحديد شكل المجموعات الضبابية LP,LN يهدف في النهاية الى اداء تصحيح كامل للضجيج في حال كان كمية مطال الضجيج عالياً .في حال وجود مطال صغير من الضجيج فإنه على العكس عملية التحسين smoothing تقلل كي تقدم حفظ افضل للصورة والاكساء . ان الضجيج المعالج يدعى ضجيج من النوع **Salt and pepper noise وهو شكل من اشكال الضجيج الذي يحدث للصورة بحيث نلاحظ ظهور بكسلات عشوائية بيضاء وسوداء. يوجد العديد من المنهجيات والتقنيات الكلاسيكية كفلتر ميديان median ، او فلاتر التشكيلي الثنائية morphological ...الخ تحدث هذه الظاهرة في حالات عندما يكون التقاط سريع للصورة**

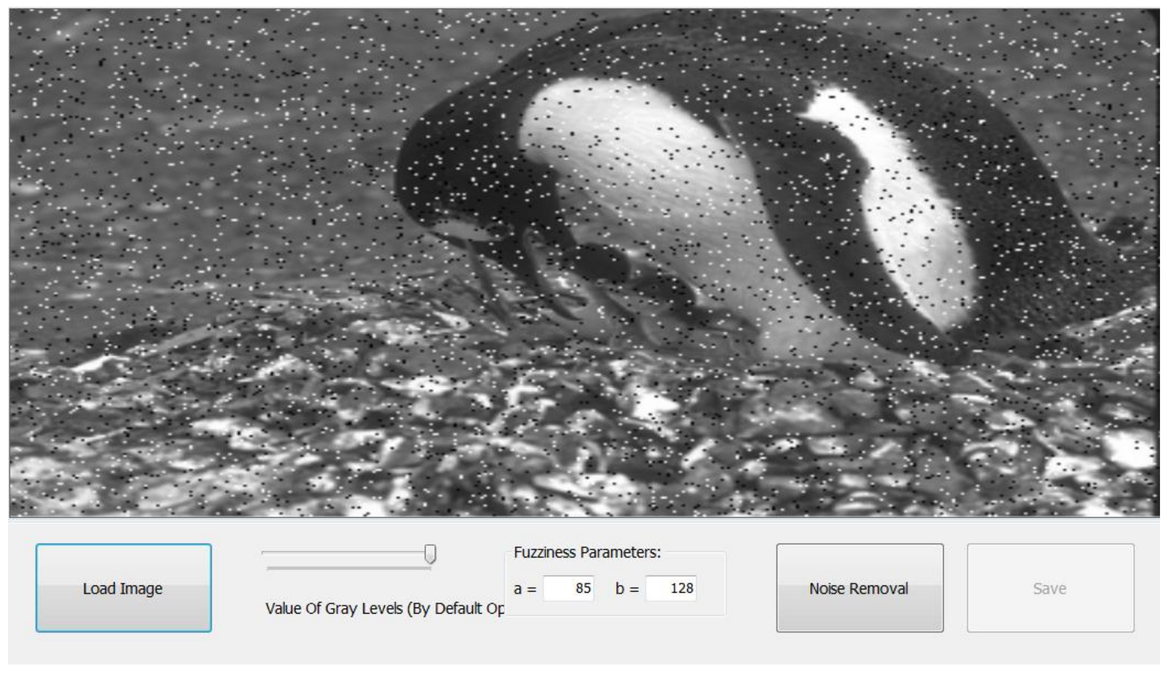
نتائج تجريبية :

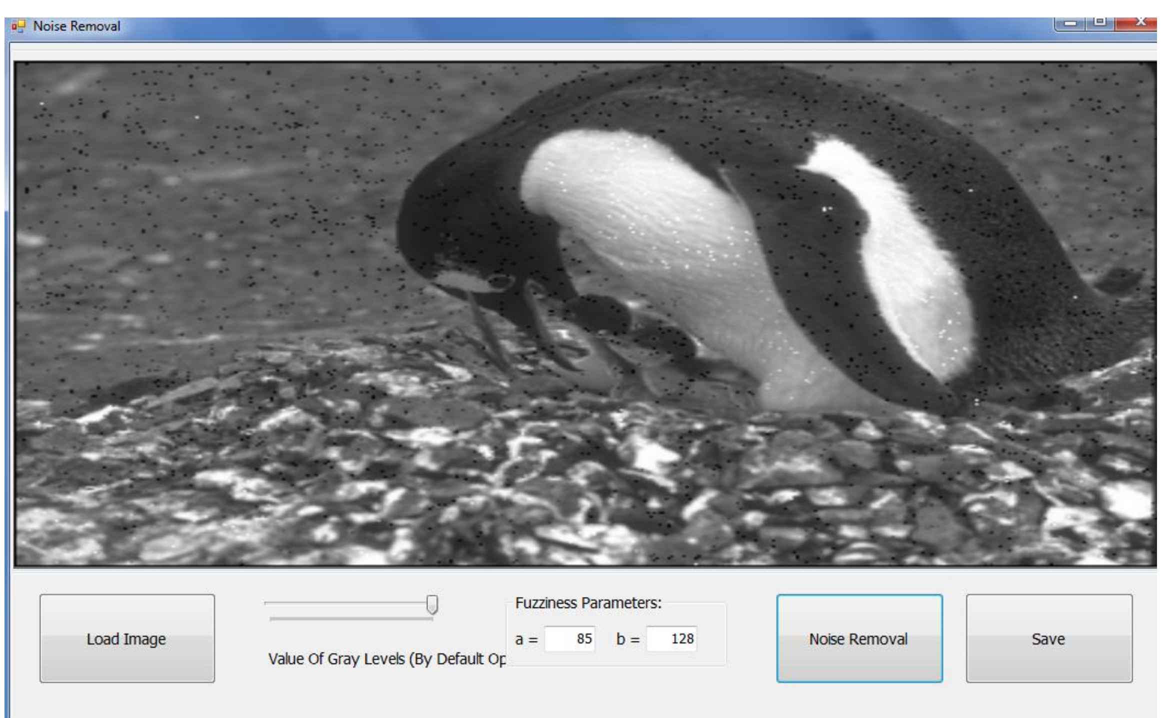


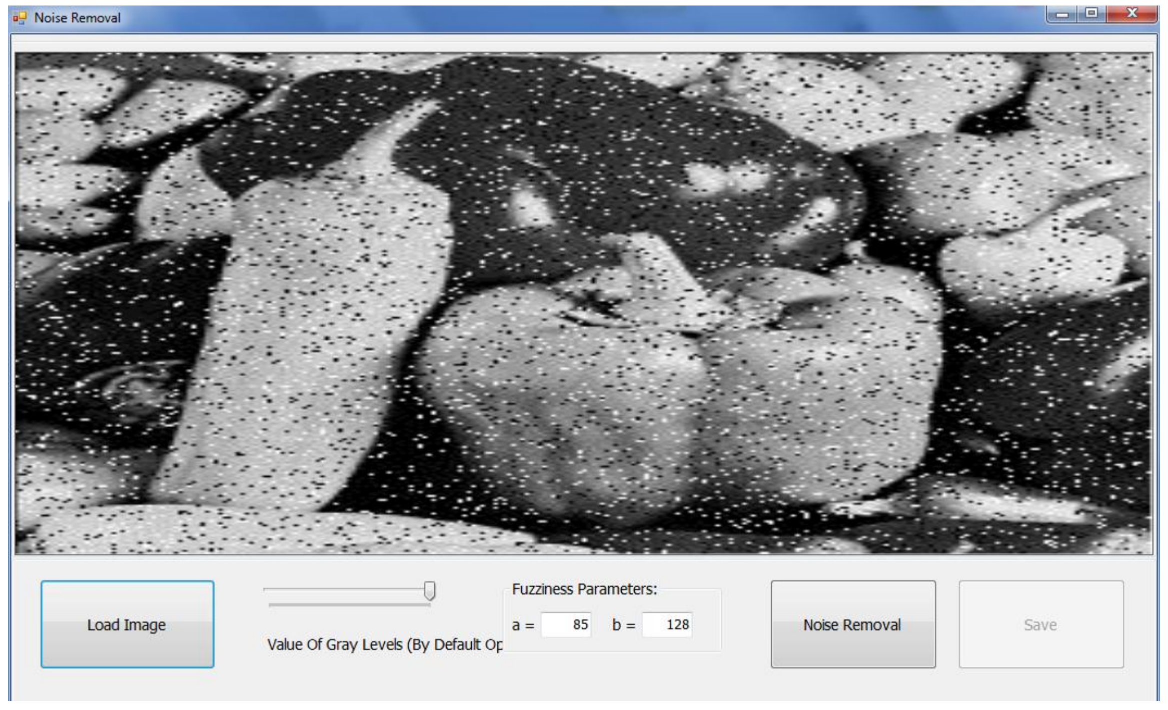



Load Image
Value Of Gray Levels (By Default Op
Fuzziness Parameters:
a = 85 b = 128
Noise Removal
Save


Load Image
Value Of Gray Levels (By Default Op
Fuzziness Parameters:
a = 85 b = 128
Noise Removal
Save


Load Image
Value Of Gray Levels (By Default Op
Fuzziness Parameters:
a = 85
b = 128
Noise Removal
Save


Noise Removal
Load Image
Value Of Gray Levels (By Default Op
Fuzziness Parameters:
a = 85
b = 128
Noise Removal
Save

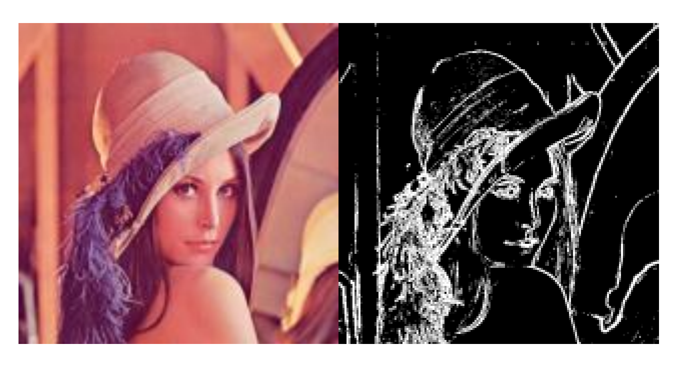

Noise Removal
Load Image
Value Of Gray Levels (By Default Op
Fuzziness Parameters:
a = 85 b = 128
Noise Removal
Save


Noise Removal
Load Image
Value Of Gray Levels (By Default Op
Fuzziness Parameters:
a = 85 b = 128
Noise Removal
Save

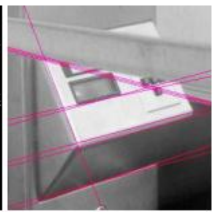
كشف الحواف باستعمال المنطق الضبابى

ماهى الحواف في الصورة الرقمية ؟

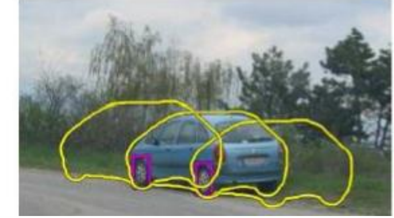
يمكن تعريف الحافة بانها الاختلاف ما بين المستويات اللونية المختلفة او بمعنى آخر هي التباين المتفاوت والكبير بين بكسل وآخر لونياً . تتضمن الحواف معلومات هامة في الصور الرقمية بحيث تقدم لنا معلومات عن مكان الغرض بحيث يفيد هذا الامر لاحقاً في تحديد الاغراض المختلفة في الصورة وملاحقتها كما في الصور المتتالية الفيديو . وكشف الحواف هو امر حيوي وهام في الرؤية الحاسبة ومعالجة الصور لأنها تستعمل في كشف خصائص الاغراض وملاحقتها ونجد اهم تطبيقات الكشف عن الحواف في الصور الطبية ، تحديد الاغراض في الفيديو والصور الثابتة ، استخراج الخلفية الخ .





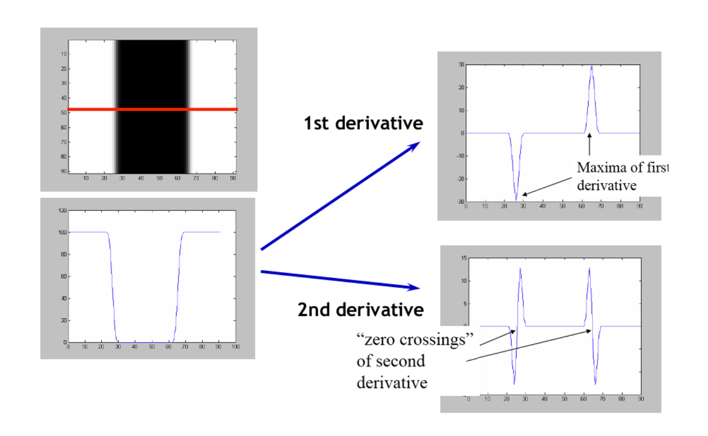






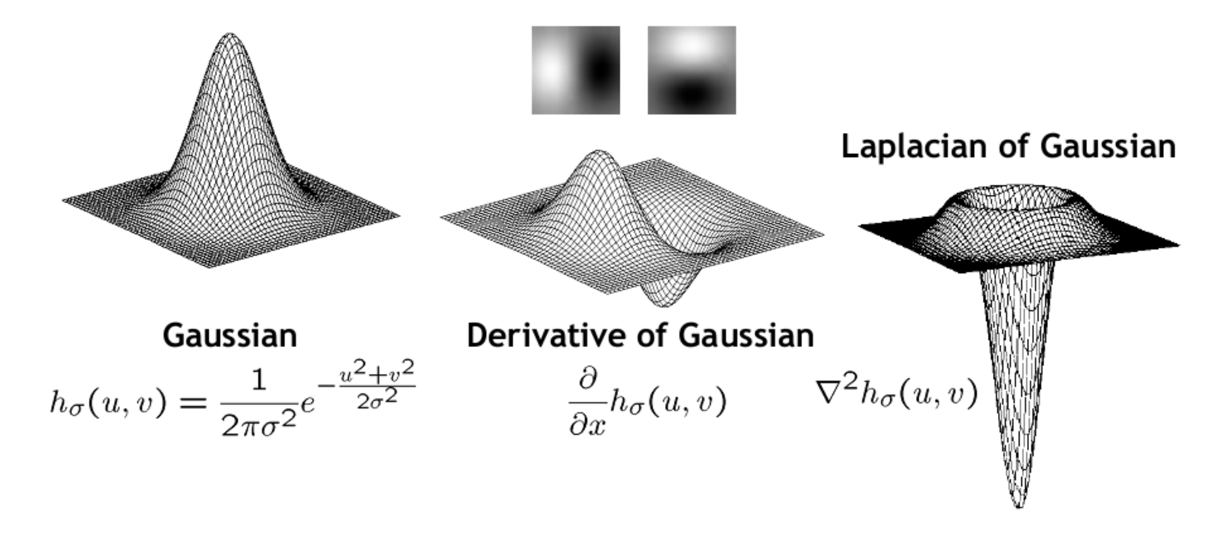

ما هو تعريف الحواف رياضياً في الصورة الرقمية ؟

بداية تعرف الصورة الرقمية على انها اشارة رقمية تحمل بكل بكسل قيمة لونية محددة . يمكن كشف الحواف من هذه الاشارة الرقمية التي تمثل الصورة عن طريق تعريف الحافة والتي هي التغيرات المفاجئة بالقيم اللونية بين بكسل واخر هذا التغير يتم التعبير عنه عند المرور الصفري بالمشتق الثاني zero crossing



ماهى الطرق المختلفة للكشف عن الحواف ؟

يوجد العديد من طرق الكشف عن الحواف واشهر الفلاتر المطبقة على الصور الرقمية هي



كما يمكن تصنيف الطرق المتبعة في كشف الحواف الى :

- طرق خطية : كمعامل سوبل Sobel ، معامل بريويت Prewitt ، معامل روبرت Robert
- طرق غير خطية : كما في الطرق التي تعتمد المنطق الضبابي FIS



معامل سوبل

$s_x =$

-1	0	1
-2	0	2
-1	0	1

$s_y =$

1	2	1
0	0	0
-1	-2	-1

معامل بريويت

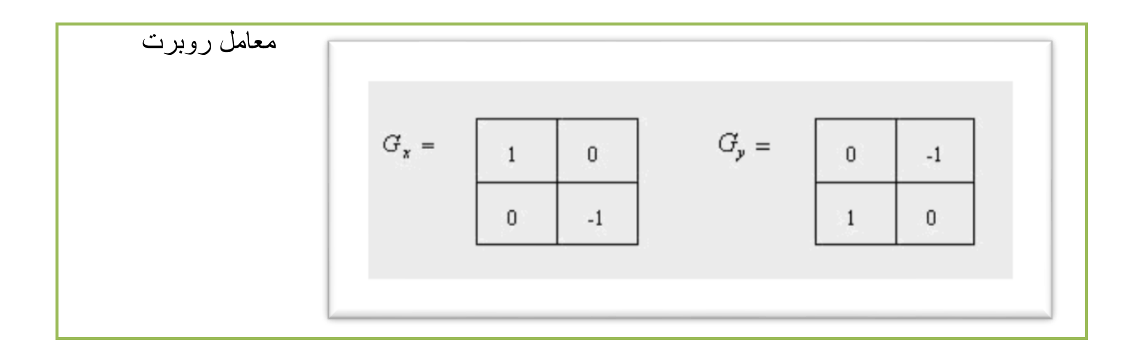
$s_x =$

-1	0	1
-1	0	1
-1	0	1

$s_y =$

1	1	1
0	0	0
-1	-1	-1

معامل روبرت



الطريقة الضبابية للكشف عن الحواف :

يمكن تمثيل الصورة الرقمية بمصفوفة ثنائية Matrix بحيث يكون كل عنصر من هذه المصفوفة يملك قيمة محددة او خاصية لكل بكسل .وخوارزميتنا تعتمد بشكل اساسي على 3 مراحل في العمل

- Image fuzzification
- Modification of membership value
- Image defuzzification

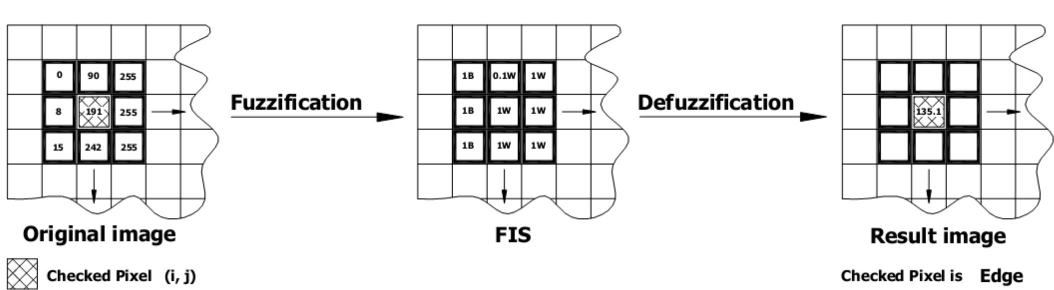
1- يتم بالمرحلة الاولى لضبضبة الصورة fuzzification
تحويل المصفوفة من شكلها النظامي " كل بكسل يملك قيمة لونية " الى الشكل الضبابي "بحيث يتملك كل بكسل فيها قيمة لدرجة انتماء هذا العنصر لمجموعة ضبابية محددة وفي مثالنا هل سيكون هذا البكسل حافة ام لا edge " .

يتم بداية العمل على تحويل الصورة من قيم اللونية rgb الى قيم التدرج الرمادي Gray scale . بحيث نجد عندها ان لكل بكسل قيمة لونية تتراوح ما بين ال 0 الى 255

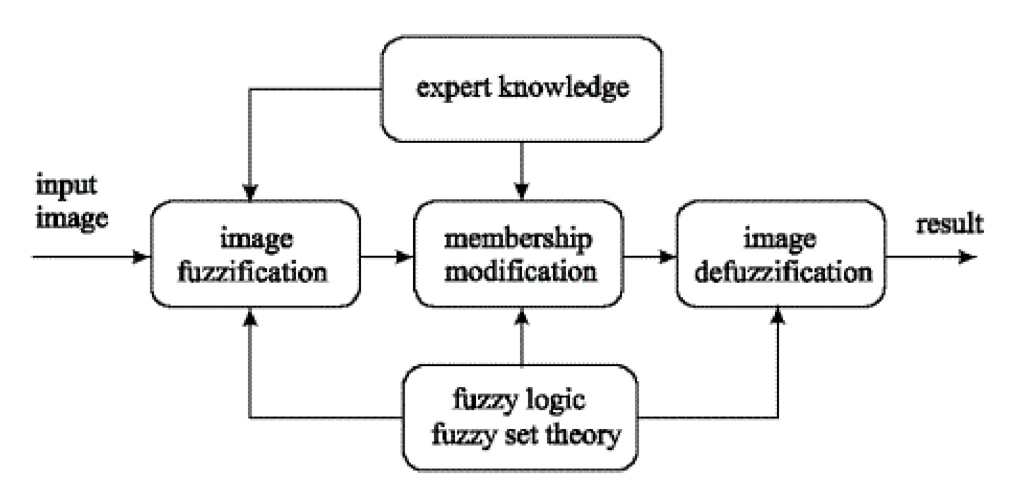
2- عملية المعالجة الضبابية :
العمليات التي تطبق على الصورة الضبابية وهذه العمليات ستؤثر على درجة الانتماء لكل بكسل . من ثم يتم تطبيق قواعد محددة لتحديد درجة انتماء البكسل المعالج ليكون حافة .

3-في المرحلة الاخيرة defuzzification يتم تحويل الصورة بطريقة معاكسة للمرحلة 1 بحيث يتم التحويل من مصفوفة قيم ضبابية الى مصفوفة تدرج بالرمادي .

ويمكن تلخيص مراحل العمل النظام الثلاث بما يلي :



نظام الاستدلال الضبابي Fuzzy Inference System (FIS)



الخطوات الاساسية لعمل النظام :

يتم ادخال الصورة ليتم تحويلها الى صورة ذات تدرج رمادي بحيث يأخذ كل بكسل قيمة ما بين 0 و 255 لتكون دخل للنظام الضبابي يتم تحويل الصورة من شكلها التقليدي الى الشكل الضبابي بحيث تملك البكسلات قيم ضبابية وذلك تبعاً لقيم تابع الانتماء المخصص للمجموعة الضبابية حافة بحيث نجد ان البكسل ذو درجة الانتماء 1 لكونه حافة سيتم مقابلته في المرحلة الاخيرة باللون الاسود الغامق ويقل تدرج الالوان كلما قلت درجة انتماء البكسل الى المجموعة الضبابية حافة .

الطريقة المقترحة للمعالجة الضبابية تفترض تقسيم الصورة الى مقاطع باستعمال مصفوفة 3*3 ثنائية .

بكسلات الحواف يتم مقابلتها بدرجة انتماءها الى المجموعة حافة . بينما تأخذ بقية البكسلات اللون الابيض .

المجموعات الضبابية :

قمنا بالعمل على تمثيل مجموعتين ضبابيتين اخريين وهما الابيض والاسود الموضحان بالشكل : كما يبين الشكل b عملية ال defuzzyfication

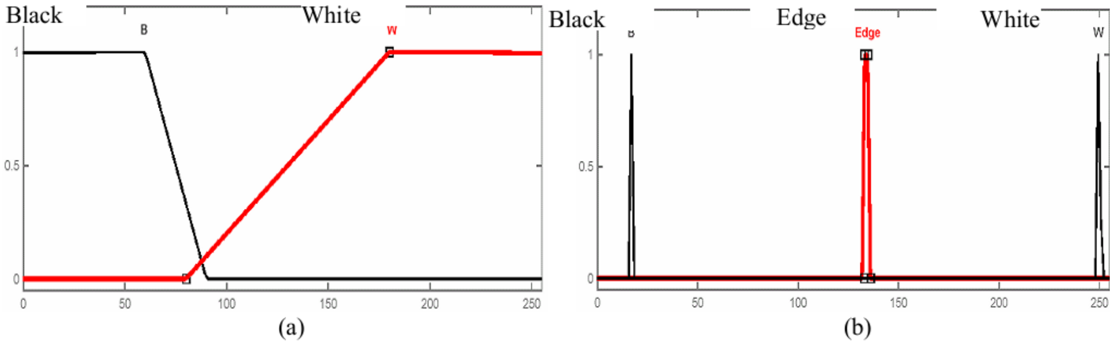
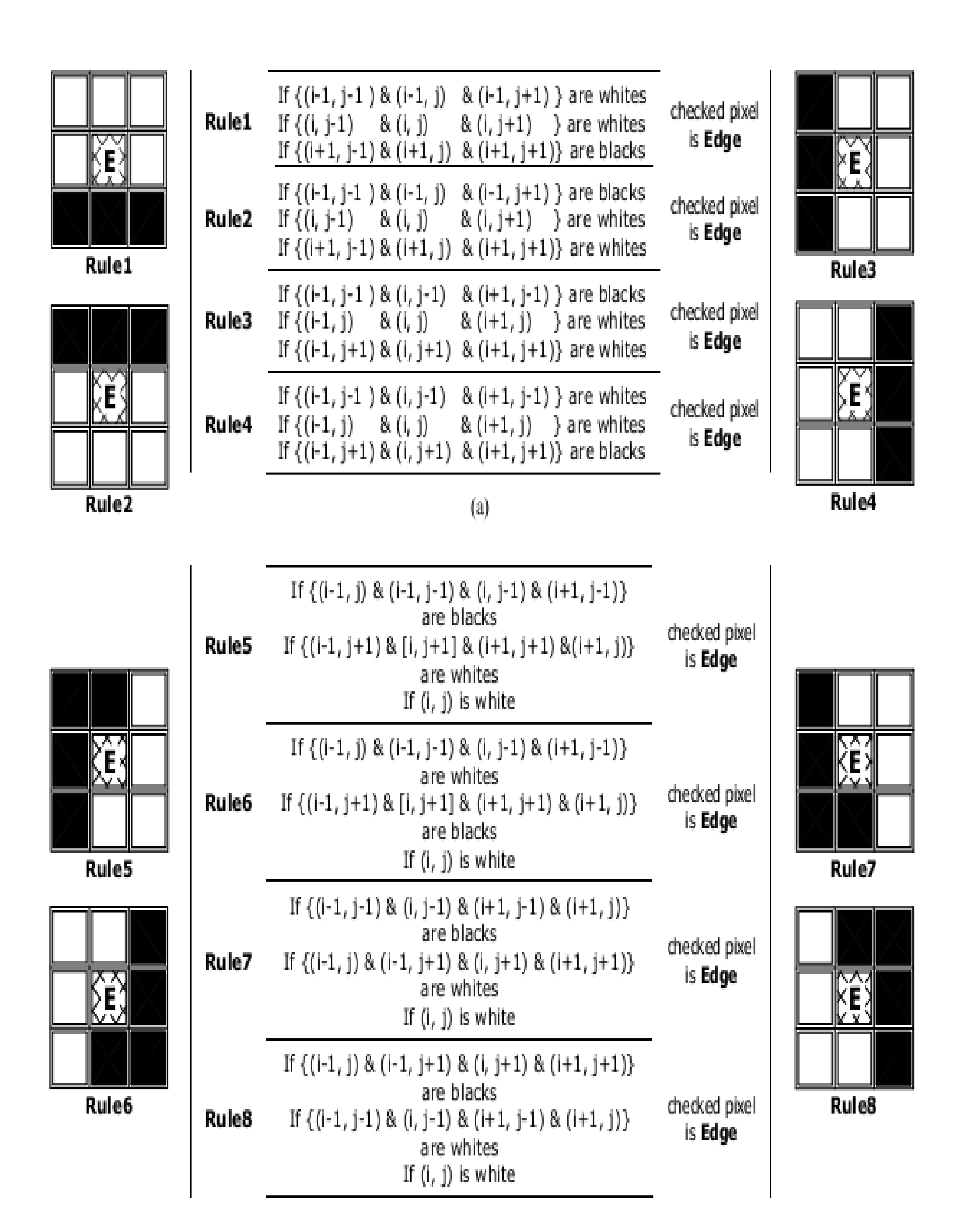


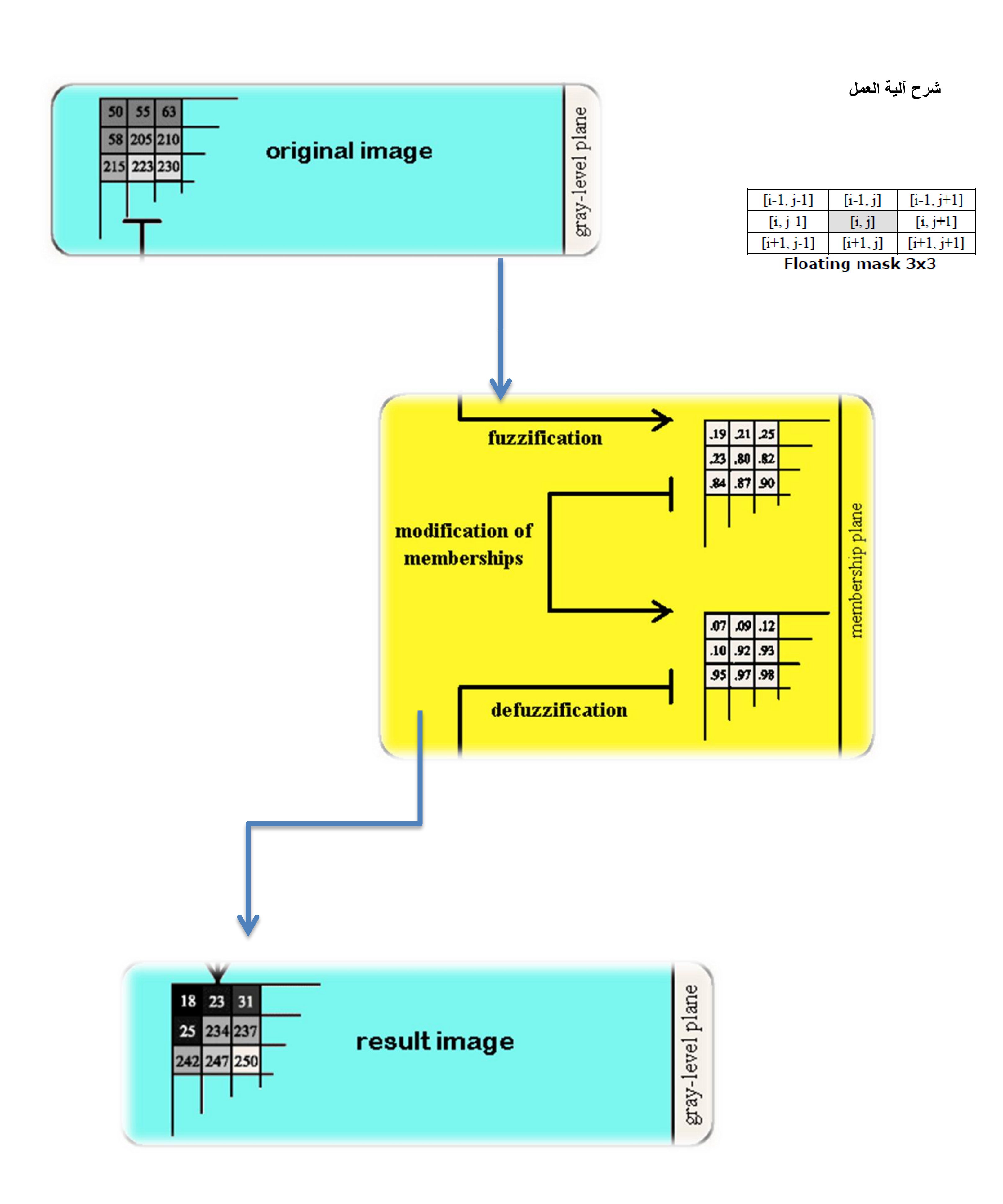
Fig. 3 Membership functions of the fuzzy sets associated to the input and to the output

بالنسبة للفلاتر والقواعد الضبابية المطبقة فهي كالتالي :

بحيث نقوم بالمقارنة بحالة الفلتر فان وافق احدى الاشكال التالية فان البكسل المعالج هو بكسل حافة ويتم حساب درجة انتماء هذا البكسل ليكون حافة فيما بعد .



(b) Fig. 4 The Fuzzy System rules



شرح آلية العمل
original image
gray-level plane
50 55 63
58 205 210
215 223 230
[i-1, j-1] [i-1, j] [i-1, j+1]
[i, j-1] [i, j] [i, j+1]
[i+1, j-1] [i+1, j] [i+1, j+1]
Floating mask 3x3
fuzzification
.19 .21 .25
.23 .80 .82
.84 .87 .90
modification of memberships
membership plane
.07 .09 .12
.10 .92 .93
.95 .97 .98
defuzzification
18 23 31
25 234 237
242 247 250
result image
gray-level plane

تسريع عمل الخوارزمية :

العديد من تطبيقات معالجة الصور تعتمد بشكل كبير على المعالجة التفرعية في العمل وذلك بسب ان الخوارزمية المطبقة تطبق على كل بكسل بنفس الخطوات ، وجاءت المعالجات الحديثة ذات الانوية المتعددة multicore بتدعيم طرق جديدة بالبرمجة بحيث اصبح بالامكان استغلال القدرة الخارقة للمعالج بالشكل الافضل .

بما ان الخوارزمية نفسها مطبقة على كل بكسل فقت قمنا بتقسم العمل الى قسمين بحيث يأخذ كل معالج قسم من الصورة ويعالجها ولقد استعملنا مكتبة TPL لما وجدنا فيها سهولة بالتعامل وقوة في الاداء .

حول مكتبة TPL :

ان مكتبة المهام المتوازية Task Parallel Library (TPL) هي جزء من مكتبة امتداد التوازي في الدوت نت ،وكما هو ظاهر من خلال الاسم فهي مبنية على مفهوم المهام .ان مصطلح توازي المهام يشير الى مهمة او اكثر مستقلة اي لايوجد ارتباط معطيات او الى ما ذلك ، بحيث يتم تنفيذ هذه المهام بشكل متوازي .ان المهمة تمثل عملية متزامنة وفي بعض الحالات هي تشبه انشاء نيسب جديد او عنصر من ساحة نياسب ThreadPool ،ولكن على مستوى اعلى من التجريد .توفر المكتبة بنى تفرعية كما في تعليمة parallel For وForEach ، باستعمال توابع معتادة و delegates.كما يمكن استعمال تلك البنى والتعليمات من خلال اي لغة برمجة تدعم الدوت نت .ان مهمة انشاء مهمات ابناء وانهاء المهام والنياسب كما ايضا مهمة اضافة عدد مهام اضافية بحسب عدد المعالجات المتوفرة كل ذلك يتم من خلال المكتبة نفسها .

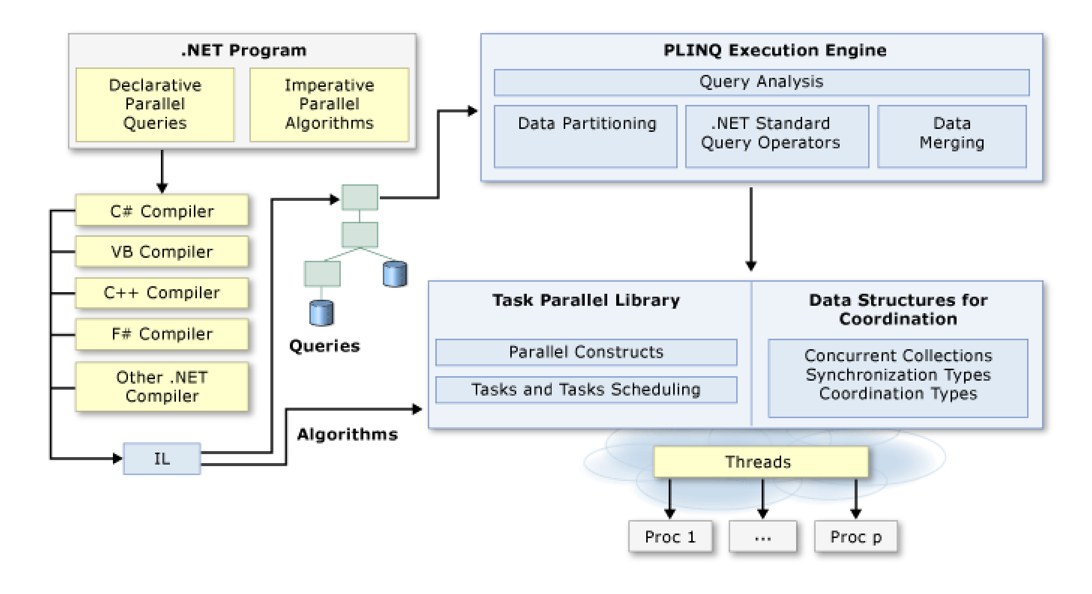
ان مكتبة TPL ايضا تضم بنى اخرى كما في Task,Future. ان البنية Task تمثل حدث يمكن تنفيذه بشكل مستقل عن باقي اجزاء البرنامج .ويمكن القول بانه يكافي النيسب ماعدا انه اقل خفة من النيسب ويحول دون انشاء نظام التشغيل لنيسب مما يؤدي الى سرعة الحركة والعمل .المهام يتم وضعها في غرض من النمط مدير المهام ضمن رتل ويتم جدولة تلك المهام لتعمل على عدة نياسب في حوض اليانسب عندما يحين دورها في التنفيذ.

البنية Future هي task التي تعيد نتيجة . النتيجة يتم حسابها في نياسب الخلفية مكبسلة بغرض من النمط future ، والنتيجة يتم وضعها في خازن مؤقت buffer لريثما يتم اعادة نتيجتها .اذا تم محاولة استرجاع النتيجة قبل ان يتم حسابها عندها فان النيسب الطالب سيتم حجبه مؤقتا ريثما تعاد النتيجة .البنية الاخرى في مكتبة TPL هي الصف Parallel توفر المكتبة TPL صيغة اساسية لبنية التفرع من خلال 3 توابع ستاتيكية في الصف Parallel.

Parallel.Invoke الذي ينفذ مصفوفة من Action delegates في التوازي ، من ثم ينتظرهم لينهي نفسه

Parallel.For مكافئة لحلقة ال for المعروفة في C#

Parallel.ForEach مكافئة لحلقة ال foreach المعروفة في C#



المعمارية :

ان المفهوم العام في مكتبة الامتداد التفرعية في الدوت نت هي المهمة ،والتي هي قسم صغير من الرماز المصدري ،عادة مايتم تمثيله كتابع لامبدا[1]، الذي يمكن تنفيذه بشكل مستقل .كلا من PLINQ و TPL تقدم منهجيات وتوابع لانشاء المهام –PLINQ تقسم الطلب query الى عدة مهام اصغر،كما ان توابع Parallel.Invoke Parallel.For, Parallel.ForEach يقسمون الحلقة الى مهام .

تتضمن المكتبة الامتدادية التفرعية PFX غرض من نمط مدير مهام الذي يقوم بجدولة المهام للتنفيذ .ان مدير المهام يحتوي على رتل من المهام عام لكل البرنامج بحيث يتم وضع المهام ضمن هذا الرتل ومن ثم يتم التنفيذ .بالاضافة الى انه يقوم بكبسلة مجموعة من النياسيب الى ماتنفذه المهام .بشكل افتراضي يتم تشكيل عدد من النياسب بعدد المعالجات المتوفرة او النوى في النظام على الرغم من ان هذا الرقم يمكن تعديله بشكل يدوي .كل نيسب يرتبط برتل خاص للنياسب والمهام .كل نيسب عندما يصبح في حالة الخمول يقوم باخذ مجموعة من المهام ويضعها في الرتل المحلي الخاص بها،من ثم ينفذها الواحدة تلو الاخرى .اذا كان الرتل العام فارغاً فان النيسب سيبحث عن المهام التي في الارتال وسياخذ المهام التي كانت في الرتل لاطول فترة .عند التنفيذ فان المهام سيتم تنفيذها بشكل مستقل مع تغيير في الحالة لكل مهمة .كنتيجة لذلك اذا كانت المهام تستعمل مصدر مشترك فيجب ان يتم مزامنة فيما بين المهام بشكل يدوي باستعمال الاقفال او السيمافورات .

إن المهام تقدم فائدتين اساسيتين :

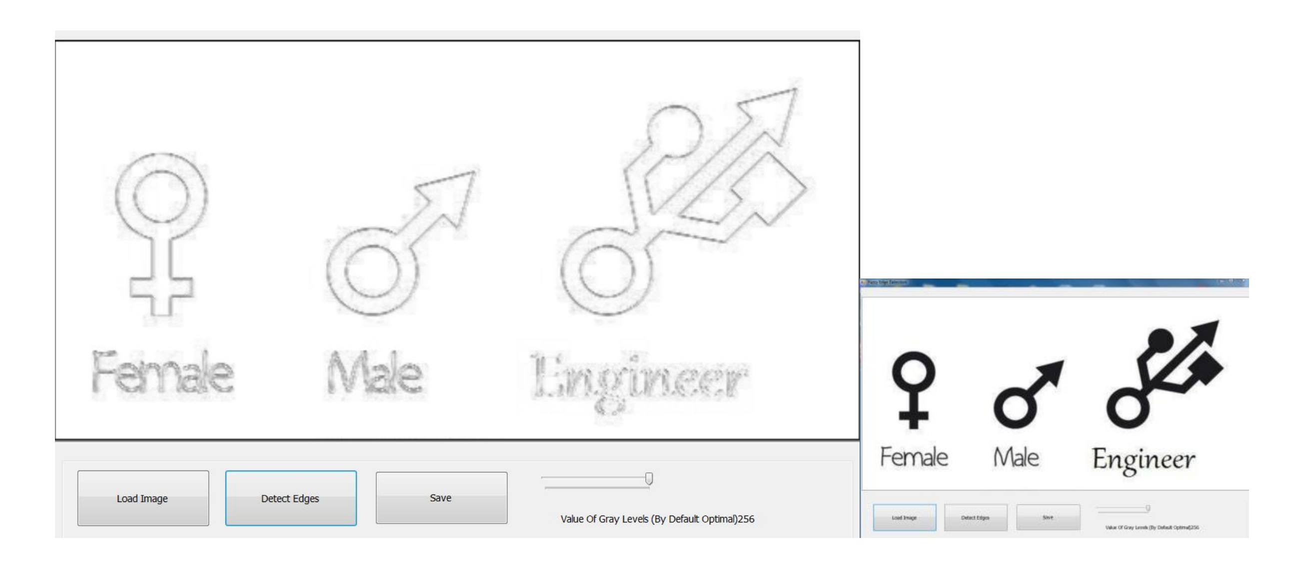
1-فعالية اكثر وامكانية استعمال الموارد بشكل اكبر .

2-تحكم برمجي اكثر من التحكم المتوفر من النياسب او عناصر العمل :

تحوي مكتبة TPL على مجموعة غنية من التوابع والمنهجيات الداعمة لعمل المهام والنياسب كتوابع الغاء المهام او الانتظار او الاستمرار او معالجة اخطاء المهام ،جدولة محددة والعديد من امور التحكم .

من اجل هذين السببين فقد تم تضمين مكتبة للتعامل مع توازي المهام وادارة عملية البرمجة التفرعية في بيئة دوت نت 4 .بحيث تم توفير API لكتابة كود متعدد النياسب ،متزامن ويعمل على التفرع مستغلا تعدد الانوية والمعالجات في الاجهزة الحديثة .

النتائج :




Fuzzy Edge Detection
Load Image
Detect Edges
Save
Value Of Gray Levels (By Default Optimal)256
Load Image
Detect Edges
Save
Value Of Gray Levels (By Default Optimal)256
Load Image
Detect Edges
Save
Value Of Gray Levels (By Default Optimal)256
Load Image
Detect Edges
Save
Value Of Gray Levels (By Default Optimal)256